

All About Drop Boxes... And What To Do When The Box Gets Dropped On You!



Verizon RISK Team – Investigating Everything

Paul Pratley
Investigations Manager – Europe Middle East & Africa
23 May 2013

PID#

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.



Just Quickly...Who We Are

The Verizon RISK Team

-Incident Response

- All Technologies + Networks
- Industrial Control Systems
- Mobile Devices

-Full Forensic Services

-Rapid Response Retainer

- In-house IR training
- Mock Incidents + Incident Readiness

-Cyber Security Intelligence

-eDiscovery



Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.



Lessons Learned...

DATA BREACH INVESTIGATIONS

- **REPORT** THE LEADING DATA SECURITY REPORT FOR SIX YEARS.
- OVER **47,000** SECURITY INCIDENTS AND **621** CONFIRMED DATA BREACH INCIDENTS.
- TURNS DATA INTO **USEFUL, ACTIONABLE INFORMATION.**



2008



2009



2010



2011



2012



DOWNLOAD THE FULL 2013 DBIR:
VERIZONENTERPRISE.COM/DBIR/2013

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

3



What Aren't Drop Boxes?



OR



Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

4



What Are They Then?



PWN Plug
\$1000



Android
Implementations
\$25-\$50



Raspberry Pi
\$35



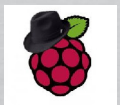
Beagle Board
\$45

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

5



Threat = "Pen Test" Distro's



PWNPI &



KALI Linux &
(Formerly Backtrack)



Debian based "Pen Testing" distro's with hundreds of tools across categories:

- Information Gathering
- IDS/IPS Identification
- Vulnerability Assessment
- Exploitation
- Privilege Escalation
- Maintaining Access
- Stress Testing



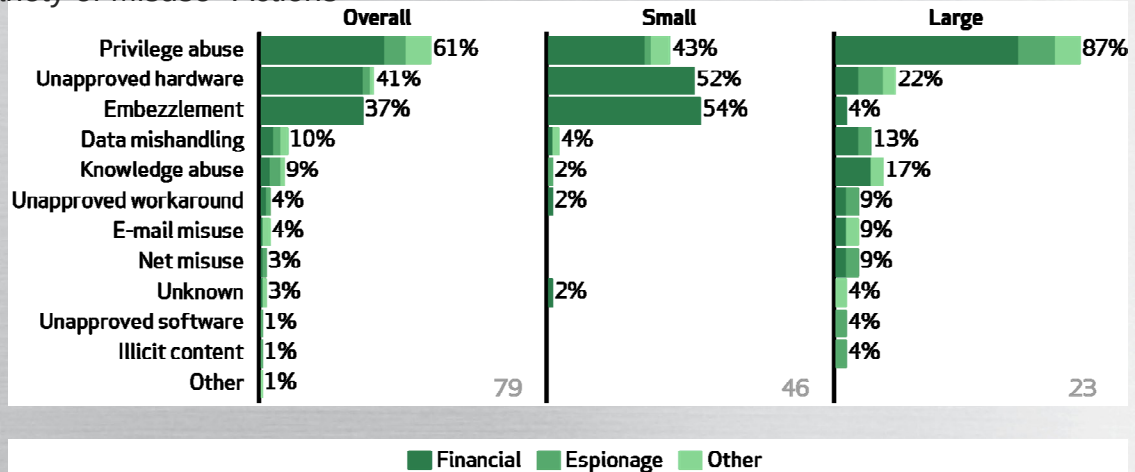
Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

6



What is the Risk?

Variety of Misuse* Actions



* Misuse accounts for 13% of Data Breaches in the 2013 DBIR

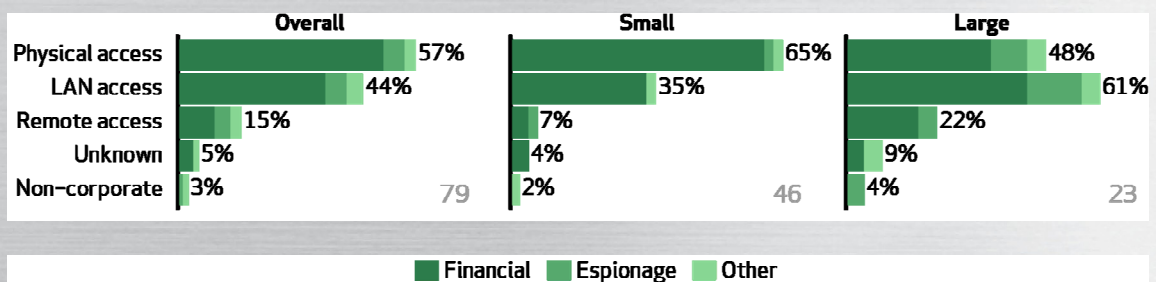
Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

7



What is the Risk?

Vector For Misuse



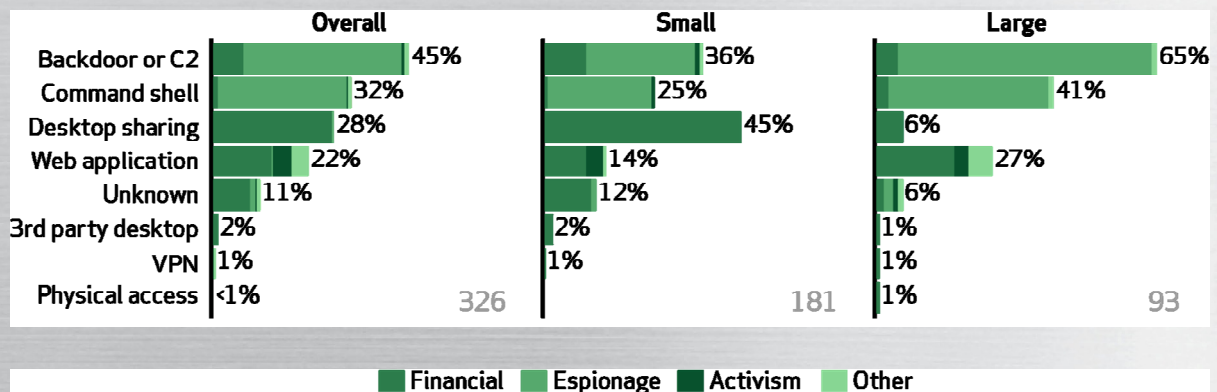
Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

8



What is the Risk?

Vector Hacking Actions - Overall



Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

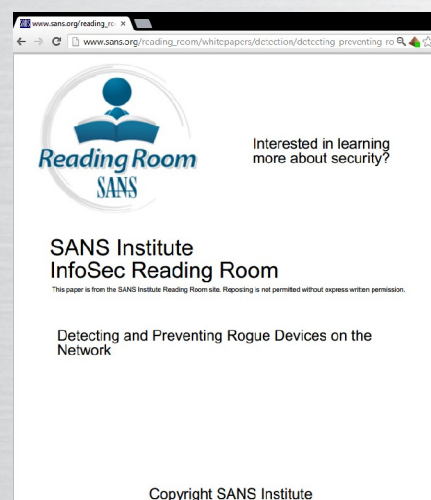
9



Do I have one on my network?

Detection Techniques:

- **Segment Networks + Security Monitoring**
 - Know your attacker, identify the highest risk assets.
 - Segment those assets.
 - Monitor and investigate unauthorized access attempts from within other network segments.
- **Deploy Rogue System Detection**
 - New devices are flagged with switch and port number for admin review.
- **Carry out physical audits prioritizing high risk areas**
 - Public areas, meeting rooms, printers, inside devices.
- **Adopt a default port-down policy**



Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

10



“Wait by the river long enough and your breach will float by”

Breach count by discovery method

Seconds				1															
Minutes		2		1															
Hours	5							2	3			1		1	2		5		
Days	3	3		4								1		5	6	1	2		
Weeks	1	5		1	10									3		1	5		
Months	1	15		16	85	9			2			1	2	4	1		2		
Years			1	1	7														
	Actor disclosure (Ext)	Fraud detection (Ext)	Monitoring service (Ext)	Customer (Ext)	Unrelated party (Ext)	Law enforcement (Ext)	Audit (Ext)	Antivirus (Int)	HIDS (Int)	NIDS (Int)	Log review (Int)	Security alarm (Int)	Fraud detection (Int)	IT audit (Int)	Report by user (Int)	Financial audit (Int)	Incident response (Int)	Unknown	Other

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

11



So you find one...now what?

Now that we are dealing with physical evidence, a whole new range of considerations come into play:

- Finger Prints
- CCTV footage
- Documentary Evidence of Contractor / Visitor Access
- Serial Numbers (**Limited manufacture and distribution**)
 - cat /proc/cpuinfo (ARM chip* serial number unique)
 - cat ifconfig (MAC address* unique)
 - SIM card ICCID (linked to identity, address and credit card)

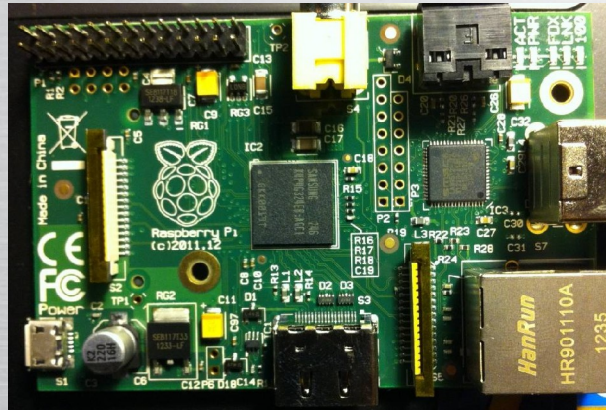
* Bear in mind that the o/s could be misrepresenting these...

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.

12

Identify The Device

- Read circuit board text
- Read chip numbers
- Identify The IP in Use
- Port / Vulnerability Scan
- Connect To It
 - HDMI
 - Composite Video
 - SSH
- Reach out to the security community



Hardware Info:	Raspberry Pi vB
O/S:	Linux
Distro:	Debian GNU/Linux 7.0 (wheezy)
Platform:	armv61
Kernel Version:	3.2.27+
Hostname:	pwnpi
IP:	10.1.2.3



Now What?

Containment

Monitoring

- TAP/Port Mirror
- PCAPs
- Border Security Devices

Get this thing off my network!

- DNS Black Hole
- Migrate
- Complete Disconnect

Preservation

Volatile Data

- System Memory
- Volatile Sys Info

Non-Volatile Data

- Use Write Blocker
- Use Forensic Boot Disk

Analysis

Volatile Data

- Volatility

Non-Volatile Data

- Std Forensic Tools

****Consider The Power Source****



History Lesson

Before:

DD /dev/mem

- Broken in newer kernels
- Memory offset issues
- Memory Size Restrictions
- Lots of context switches and memory loss due to overwriting free pages

```
root@pwnpi:/# cat /proc/iomem
00000000-1effffff : System RAM
  00008000-004c0e77 : Kernel text
  004e2000-005b5127 : Kernel data
20000000-20000fff : bcm2708_vcio
20003000-20003fff :
bcm2708_systemtimer
20006000-20006fff : bcm2708_usb
  20006000-20006fff : dwc_otg
20007000-20007fff : bcm2708_dma.0
```

LiME – Linux Memory Acquisition

- First announced at ShmooCon2012
- Loadable Kernel Module (LKM)
- Operates only in the kernel
- Widely Supported
 - Typical *nix support
 - Arm Support
 - Android Support
- Small Memory Footprint

lime-forensics
LiME - Linux Memory Extractor

Project Home Downloads Issues Source

Search Current downloads for Search

Filename	Summary + Labels	Uploaded	ReleaseDate	Size	DownloadCount
LIME_Documentation_1.1.pdf	LiME Forensics Documentation	Mar 19	Mar 19	629 KB	799
lime-forensics-1.1-r17.tar.gz	LiME Forensics Source + Documentation	Mar 19	Mar 19	597 KB	645

1 - 2 of 2

code.google.com/p/lime-forensics/

- **You need to compile a LiME binary for your memory acquisition**
 - Virtualise* Pentest O/S and Compile
 - Virtualise* same Kernel / Architecture
 - Buy / Borrow / Steal same device and compile on physical device
- **Future Possibility**
 - DD the SD Card and virtualise using LiveView
 - vPi project (VMWare Virtualisation)

*Requires QEMU ARM Emulator

QEMU - Emulating Raspbian

xeccdesign.com/qemu-emulating-raspbian-pi-the-easy-way/

XEC DESIGN

MOSTLY TUTORIALS FOR RASPBERRY PI

HOME

Home • QEMU - Emulating Raspberry Pi the easy way (Linux or Windows)

QEMU – Emulating Raspberry Pi the easy way (Linux or Windows!)

This page will show how to emulate an ARM1176 system the quick and easy way.

UPDATE (17th of March, 2013): In order to run the latest image (2013-02-09-wheezy-raspbian.img) you need to comment out the contents of /etc/ld.so.preload. - kinsaa

Linux users will find the necessary steps [here](#).

I have created a patcher for windows users and it's available [here](#). When asked, give it the directory containing your 2013-02-09-wheezy-raspbian.img file. The patched img file should be properly bootable by qemu.

Assumptions

You have:

- QEMU - Linux or Windows. Make sure you have a suitable version (see 'Quick note on QEMU and ARM1176').
- A disk image for your distro of choice
 - Raspian and Debian should work out of the box, however Arch Linux requires a few tweaks to work. These will not be explored in this article, as they can only be done in Linux.
- Ability to read and follow instructions carefully

Quick note on QEMU and ARM1176

ARM1176 support is relatively new. If you did not compile QEMU yourself from git, or download fairly recent binaries, replace -cpu arm1176 with -cpu arm1136-r2 whenever you see it. Note that you will be missing out on many important bug fixes and a few unimportant CPU features. If you boot with -cpu arm1176 and see something about unsupported instructions, you've got an older version of QEMU.



Totto, we're not in x86 land any more!!

Download the correct Kernel Headers (ie for PWNPI 3.2.27+)

```
$ cd /usr/src
```

```
$ wget http://repo.anconafamily.com/repos/apt/raspbian/pool/main/l/linux-upstream/linux-headers-3.2.27+_3.2.27+-3_armhf.deb
```

```
$ dpkg -i linux-headers-3.2.27+_3.2.27+-3_armhf.deb
```

- SymLink `/lib/modules/3.2.27+/build` to `/usr/src/linux-headers-3.2.27+`
- Compile LiME



Path

- Either a path "`<path>`"(der) or port for listening and pushing the memory out to "`tcp:<port>`"

Format

- RAW – Cats segments together
- Padded – Inserts Zeros between memory segments
- Lime – Integrates address space range for each segment into a header (best for Volatility)

DIO – Direct IO

- Bypasses kernel to write directly to media (does this by default anyhow)



Network Acquisition

- Copy locally (Win SCP)
- Execute on Pi: `# insmod <path>/lime.ko "path=tcp:666 format=lime"`
- Collect on Workstation: `$ nc <Pi IP Add> 4444 > Pi_Memory.lime`

Local Acquisition

- Copy to USB Flash
- Execute LiME: `# insmod <path>/lime.ko "path=<path> format=lime"`



DEMO TIME



Android Debug Bridge (ADB)

•Put the device into USB Debug Mode

- Sometimes Requires special cables
- Can be a problem if security policies have disabled USB debug mode
- Can require reboot (pointless)
- Use a USB flash drive, write to USB
- Acquire SD card and then copy lime to the SD card and write memory to the card

```
$ adb push <path>lime.ko /sdcard/lime.ko
```

```
$ adb forward tcp:666 tcp:666
```

```
$ adb shell
```



Collect Other Volatile Data:

- Uptime - Great intel as to when attacker installed the device, correlate with:
 - CCTV
 - Employee access card logs
 - Keysafe Logs
 - Contractor / Visitor Logs
- Date – Determine accuracy of system clock
- Netstat -nao

Unplug and Image SD card or DD in place



Analysis is relatively straight forward

- Linux memory analysis in Volatility Framework
 - Need to create a profile for each device
 - apt-get install dwarfdump (and GCC/make + Kernel headers)
 - Check out the volatility source code
 - Make Dwarffile
 - \$ cd volatility/tools/linux
 - \$ make
 - \$ head module.dwarf
 - Get the system.map file (/boot)
 - Place both module.dwarf and system.map into a zip file....now you have your profile



Things you can do:

- PSList – List all processes and offsets
- PSTree – List the parent / child relationships (ie should see bash spawned from ssh)
- PSaux – Process arguments
- Proc_maps – map out process memory space
- Dump_map – get the binary and the static data (great for binary reversing)
- Kernel objects, Debug Buffer, Kernel memory caches
- Recover APP Table, ifconfig, routing cache, netstat output, per-socket packet queues



Disk Analysis

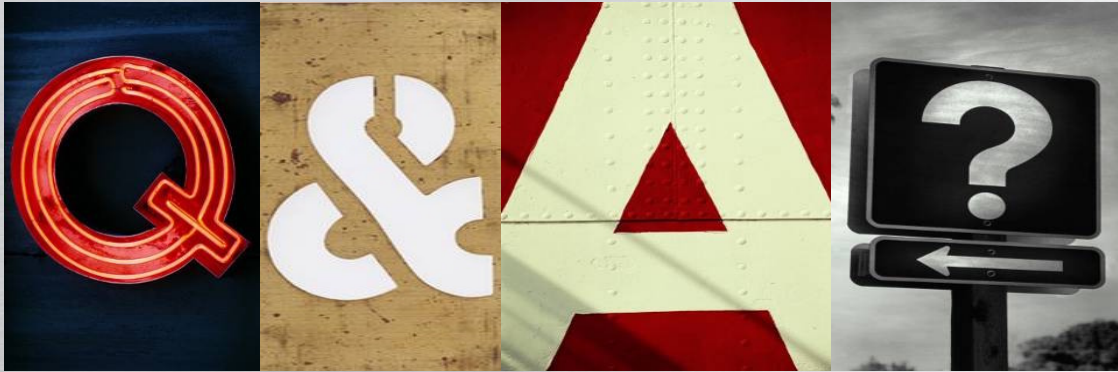
- Disk analysis in your tool of choice (Open Source / EnCase / FTK)
 - Hash all files in Distro, create a filter
 - GREP for IPs
 - Timeline Analysis
 - Reverse any interesting Binaries



Don't forget your other big problem...

You've only discovered one slice of the Pi





Verizon RISK Team

In case of an incident, contact us 24/7 worldwide:

Phone: +1.877.330.0465

Email: ir-global@verizon.com

Confidential and proprietary materials for authorized Verizon personnel and outside agencies only. Use, disclosure or distribution of this material is not permitted to any unauthorized persons or third parties except by written agreement.